Санкт-Петербургский Государственный Электротехнический Университет "ЛЭТИ"

кафедра физики

Задание №1 по дисциплине

"Физические основы информационных технологий"

Название: Искривление луча в оптическом канале

|  |  |
| --- | --- |
| Фамилия И.О.: | Сырцева Д.Д |
| группа: | 1303 |
| Преподаватель: | Альтмарк А.М. |
| Итоговый балл: |  |
|  |  |

Крайний срок сдачи: 22.10.23

.

Санкт-Петербург 2023

**Условие задания**

Найти длину траектории светового луча *S* в прямолинейном дисперсионном оптоволоконном канале, Рис.1, c показателем преломления n1. Оптоволокно окружено средой с показателем преломления n2. Функцию распределения показателя преломления n1(y, ω) можно представить как:

,

где y – поперечная координата, ω – циклическая частота светового луча.

Функцию , функцию Zf(y), описывающую координату z выходного торца волновода, начальный угол ввода луча α в волновод, координату ввода луча в волновод *y*=y0, радиус канала R можно взять в файле FOIT\_IDZ1.xlsx. Все геометрические размеры даются в безразмерных координатах.

Необходимо построить график траектории луча, а также записать ответ *S* в текстовый файл IDZ1\IDZ1.txt. Помимо текстового файла IDZ1.txt в папке IDZ1 должен находиться Word-файл (Pdf-файл) с отчетом, а также файл с кодом (Python, Mathcad, Mathematica). Для лучшего понимания отчетности смотрите папку “Пример организации яндекс-папки студентов”.

Пример содержания файла IDZ1.txt:

4.53258

n2

Y

R

Zf(y)

y0

n1

Z

α

0

n2

-R

n2

Рисунок.1

**Исходные данные (11 вариант)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| R | n2 | f1(y) | Zf(y) | ω \* 10^14, рад/с | y0 | α , град |
| 0.8 | 1 | 1.4 - 0.12\*Cos[5\*y] | 28 + 3\*Sin[17.951958020513104\*y] | 3.3 | 0.2 | 20 |

**Основные теоретические положения**

Закон преломления описывает изменение направления распространения света (или других электромагнитных волн) при переходе из одной среды в другую. Этот закон устанавливает связь между углами падения и преломления света и показателями преломления двух сред, через которые свет проходит. Вот его математическое выражение:

n₁ \* sin(θ₁) = n₂ \* sin(θ₂)

* n₁ и n₂ - показатели преломления первой и второй среды соответственно.
* θ₁ - угол падения света на границу между двумя средами.
* θ₂ - угол преломления света во второй среде.

Показатель преломления - это величина, характеризующая, насколько свет замедляется при прохождении через данную среду по сравнению с его скоростью в вакууме. Каждая среда имеет свой собственный показатель преломления, который может быть различным для разных типов материалов.

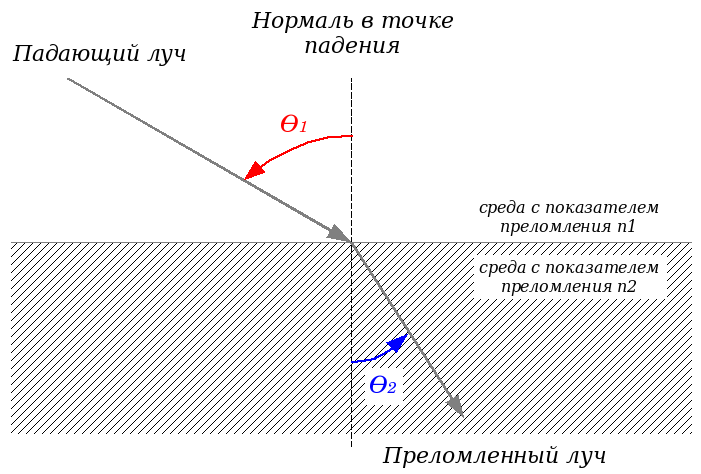


Рис1. Преломление луча

Полное внутреннее отражение - это оптическое явление, при котором световая волна, переходя из одной оптически более плотной среды в менее плотную, не покидает границу между этими средами, а полностью отражается обратно в более плотную среду. Это явление происходит при определенных условиях:

* Свет должен попадать из оптически более плотной среды в менее плотную.
* Угол падения света должен превышать критический угол. Критический угол зависит от показателей преломления двух сред и определяется следующим образом:

sin(θкр) = n₂ / n₁

* n₁ и n₂ - показатели преломления более и менее плотной среды соответственно.
* θкр – критический угол.

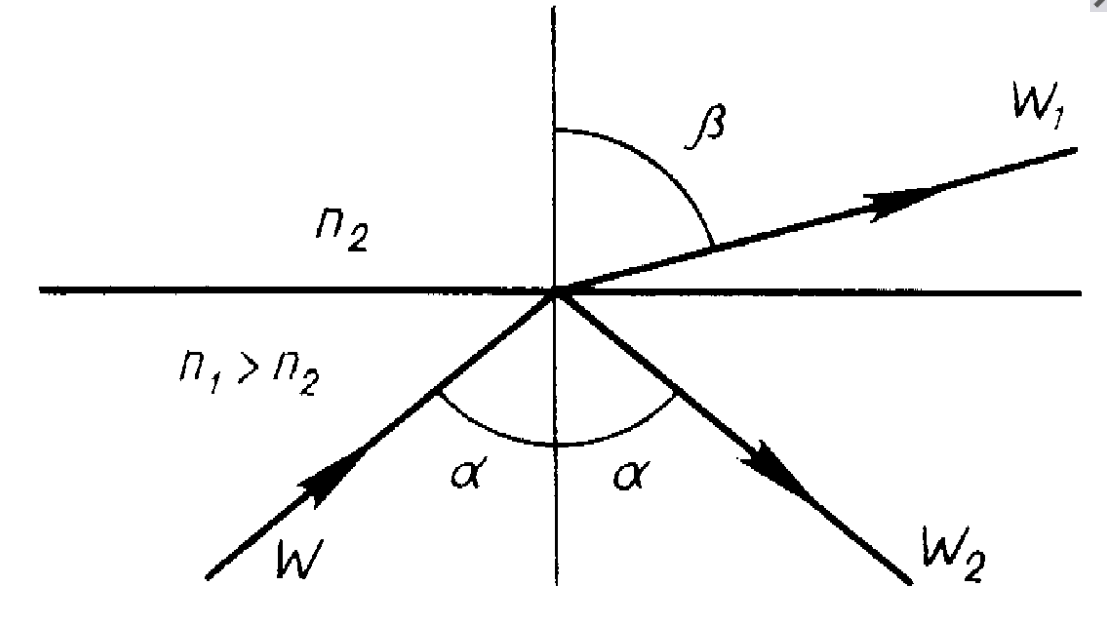


Рис 2. Полное внутреннее отражение

Полное внутреннее отражение играет важную роль в плоских оптических волноводах. Волноводы используют этот феномен, чтобы удерживать свет внутри своей структуры и направлять его так, чтобы он следовал определенным путям.

**Вывод:**

В процессе выполнения индивидуального домашнего задания стало ясно, что полное внутреннее отражение, так же, как и закон Снеллиуса является важным явлением при прохождении луча в волноводе, также в результате удалось найти длину траектории луча.

**ПРИЛОЖЕНИЕ А**

**ИСХОДНЫЙ КОД ПРОГРАММЫ**

Файл **main.py**

import numpy as np

import matplotlib.pyplot as plt

def f1(y):

return 1.4 - 0.12 \* np.cos(5 \* y)

def z\_f(y):

return 28 + 3 \* np.sin(17.951958020513104 \* y)

def n1(y):

return f1(y) \* (1 - ((0.35 \* 10 \*\* 14) / (3.3 \* (10 \*\* 14))) \*\* 2)

def plot\_trajectory(y\_array, z\_array, r):

y = np.arange(-r, r, 0.001)

plt.plot(z\_array, y\_array, 'r', label='График траектории луча')

plt.plot(z\_f(y), y, 'b', label='График выходного торца волновода')

plt.axhline(r, color='b', linestyle='--')

plt.axhline(-r, color='b', linestyle='--')

plt.axvline(0, color='b', linestyle='--')

plt.xlabel('Z координата')

plt.ylabel('Y координата')

plt.legend()

plt.grid()

plt.show()

def search(a0, n0, r, part, y0, z0):

a0 = np.deg2rad(a0)

result\_len = 0

direction = 1

y\_array = []

z\_array = []

y\_array.append(y0)

z\_array.append(z0)

b = np.arcsin((np.sin(a0) \* n0) / n1(y0))

y\_array.append(y0 + np.sin(b) \* part)

z\_array.append(0 + np.cos(b) \* part)

n\_2 = n1(y\_array[-1])

n\_1 = n1(y0)

a = np.pi / 2 - b

b = np.arcsin(np.clip((np.sin(a) \* n\_1) / n\_2, -1, 1))

result\_len += part

while z\_array[-1] <= z\_f(y\_array[-1]):

if 1 == abs(np.sin(b)):

b = a

direction \*= -1

y\_array.append(y\_array[-1] + np.cos(b) \* part \* direction)

z\_array.append(z\_array[-1] + np.sin(b) \* part)

if abs(y\_array[-1]) <= r:

n\_2 = n1(y\_array[-1])

else:

n\_2 = n0

n\_1 = n1(y\_array[-2])

a = b

b = np.arcsin(np.clip((np.sin(a) \* n\_1) / n\_2, -1, 1))

result\_len += part

return result\_len, y\_array, z\_array

def main():

a0 = 20

n0 = 1

r = 0.8

y0 = 0.2

z0 = 0

one\_part = 0.0001

result\_len, y\_array, z\_array = search(a0, n0, r, one\_part, y0, z0)

print(f"Результирующая длина траектории = {result\_len}")

plot\_trajectory(y\_array, z\_array, r)

if \_\_name\_\_ == "\_\_main\_\_":

main()